



Science Arts & Métiers (SAM)

is an open access repository that collects the work of Arts et Métiers Institute of Technology researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: <https://sam.ensam.eu>
Handle ID: <http://hdl.handle.net/10985/17818>

To cite this version :

Luis RAMÍREZ, Charles FOULQUIÉ, Xesús NOGUEIRA, Sofiane KHELLADI, Jean-Camille CHASSAING, Ignasi COLOMINAS - New high-resolution-preserving sliding mesh techniques for higher-order finite volume schemes - Computers & Fluids - Vol. 118, p.114-130 - 2015

Any correspondence concerning this service should be sent to the repository

Administrator : scienceouverte@ensam.eu



New high-resolution-preserving sliding mesh techniques for higher-order finite volume schemes

Luis Ramírez^a, Charles Foulquié^b, Xesús Nogueira^{a,*}, Sofiane Khelladi^b, Jean-Camille Chassaing^c, Ignasi Colominas^a

^a Grupo de Métodos Numéricos en Ingeniería, GMNI, Universidade da Coruña, A Coruña, Spain

^b Laboratoire de Dynamique des Fluides, Arts et Métiers ParisTech, 151 boulevard de l'Hôpital, 75013 Paris, France

^c Sorbonne Universités, UPMC Univ Paris 06, CNRS, UMR7190, D'Alembert Institute, F-75005 Paris, France

A B S T R A C T

This paper presents a new sliding mesh technique for the computation of unsteady viscous flows in the presence of rotating bodies. The compressible Euler and incompressible Navier–Stokes equations are solved using a higher-order (>2) finite volume method on unstructured grids. A sliding mesh approach is employed at the interface between computational grids in relative motion. In order to prevent loss of accuracy, two distinct families of higher-order sliding mesh interfaces are developed. These approaches fit naturally in a high-order finite volume framework. To this end, Moving Least Squares (MLS) approximants are used for the transmission of the information from one grid to another. A particular attention is paid for the study of the accuracy and conservation properties of the numerical scheme for static and rotating grids. The capabilities of the present solver to compute complex unsteady vortical flow motions created by rotating geometries are illustrated on a cross-flow configuration.

1. Introduction

Flow unsteadiness and noise generation are currently among the most important limitations for the design of turbomachinery and renewable energy devices. These configurations involve complex unsteady flow patterns like vortex shedding, stalled flows, blade wake interactions which are, in general, due to the presence of moving or oscillating bodies. On one hand, one must employ high-order numerical methods to accurately compute both the unsteady flow field and the aeroacoustic field. On the other hand, dedicated techniques must be employed to carefully deal with the interface between static and moving grids in an unsteady flow framework. Such issue can be addressed using several numerical approaches, among others, the phase-lagged periodic boundary conditions for rotor–stator interaction in axial compressor [10,16,17], the body-fitted approach in an Arbitrary Lagrangian Eulerian (ALE) setting, Cartesian grid methods based on the immersed-boundary [40] or on the cut cell methods [3,54] and the non-boundary conforming sliding mesh approach. The later is attractive due to its ability to capture flow unsteadiness without requiring the use of a filtering procedure nor computationally

expensive re-meshing strategies. The sliding mesh method was successfully employed by Rai [42,43] for the computation of rotor–stator interactions in a supersonic flow. This patched-grid technique allows relative sliding of one mesh adjacent to another static or moving mesh. A three steps explicit zonal scheme, which preserve flux conservation at the interface, is proposed in [41]. More recently, Gourdain et al. [15] employed the sliding mesh approach for the simulation of large-scale industrial flows in multistage compressors. In a comparative study between Chimera and sliding mesh techniques for unsteady simulations of counter rotating open-rotors, Francois et al. [14] shown that these methods give similar accuracy but the later requires much less memory than the Chimera approach. Note also that the sliding mesh algorithm was used by Steijl and Barakos [46] for the computational fluid dynamic analysis of helicopter rotor-fuselage aerodynamics.

Nowadays, sliding mesh techniques are commonly used to compute non-axisymmetrical unsteady flow fields and corresponding aerodynamic performances of cross-flow fans [33,49] and wind turbines [18,20,21,23,26,19,22,1]. In particular, McNaughton et al. [31] obtained a good agreement between coupled LES-sliding interfaces for thrust and power predictions of a tidal-stream turbine. As far as aeroacoustic computations are concerned, Moon et al. [33] developed a time-accurate viscous flow solver for the prediction of unsteady flow characteristics and the

* Corresponding author.

E-mail address: xnogueira@udc.es (X. Nogueira).

associated aeroacoustic blade tonal noise of a cross flow impeller. The sliding mesh approach, which was implemented in an unstructured finite-volume solver on triangular meshes, was able to correctly predict the oscillations of the eccentric vortex due to the mismatch of blade incidence angles and the recirculation bubbles around the blades.

However, most of the sliding mesh methods proposed in the literature belong to the family of low order interpolation schemes. Therefore they cannot be used in conjunction with higher-order numerical schemes without depreciating the overall accuracy of the numerical methods.

To the authors knowledge, few studies addressed such problem. A high order ($order \geq 3$) h/p Discontinuous Galerkin method with sliding mesh capabilities was recently proposed by Ferrer and Willden [12] for the computation of the unsteady incompressible flow field of a three bladed cross-flow turbine. They have successfully obtained spectral convergence rate when solving the incompressible Navier–Stokes equations on non-conformal grids. In [2] a mesh moving technique for sliding interfaces is presented for the numerical simulation of a wind turbine with a FEM-based ALE-VMS (variational multiscale formulation written in the arbitrary Lagrangian–Eulerian frame) formulation.

In this work, we intent to develop higher-order sliding mesh interface for the solution of transient flows on mixed rotating and static computational domains. To this end, we consider a high-resolution finite volume method based on Moving Least Squares (MLS) reconstructions.

The theoretical fundamentals of the used finite volume method (FV-MLS) were presented in [9,24,36,35,44] and references therein. A first application of FV-MLS for turbomachinery aeroacoustics was presented in [38]. In those works, artificial acoustic sources were propagated using the Linearized Euler Equations. Only stator blades and rotating sources into the propagating medium were considered. This first tentative permits to study the attenuation due to the acoustic screen effect of stator blades. The next step is to introduce the rotating part into the propagation medium by the use of sliding mesh method coupled to FV-MLS solver. In this work we present a sliding mesh model based on the use of Moving Least Squares (MLS) approximants [25]. It is used with a high-order (>2) finite volume method that computes the derivatives of the Taylor reconstruction inside each control volume using MLS approximants [9,24,36,35]. Thus, this new sliding mesh model fits naturally in a high-order finite volume framework for the computation of acoustic wave propagation into turbomachinery. We present two different approaches based on MLS approximants for the transmission of information from one grid to another. An interface-type sliding mesh approach, and a new methodology that does not require the computation of intersections.

The paper is organized as follows. In Section 2 the governing equations are written. In Section 3, the basic finite volume formulation is presented. Moving Least Squares (MLS) approximation and the FV-MLS method are briefly described in Section 4. The new MLS-based sliding-mesh technique is presented in Section 5. Then, Section 6 is devoted to numerical simulations. Finally, the conclusions are drawn.

2. Governing equations and numerical methods

In order to account the relative mesh motion of one mesh with respect to other, it is advantageous to write the two dimensional compressible Navier–Stokes equations in the Arbitrary Lagrangian–Eulerian (ALE) form,

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial (\mathbf{F}_x - \mathbf{F}_x^V)}{\partial x} + \frac{\partial (\mathbf{F}_y - \mathbf{F}_y^V)}{\partial y} = \mathbf{0} \quad (1)$$

where \mathbf{U} is the vector of variables $\mathbf{F} = (\mathbf{F}_x, \mathbf{F}_y)$ is the inviscid flux vector and $\mathbf{F}^V = (\mathbf{F}_x^V, \mathbf{F}_y^V)$ is the viscous flux vector.

For compressible flows the conservatives variables are defined as

$$\mathbf{U}(\mathbf{x}, t) = \begin{Bmatrix} \rho \\ \rho v_x \\ \rho v_y \\ \rho E \end{Bmatrix} \quad (2)$$

and the inviscid fluxes are given by

$$\mathbf{F}_x = \begin{pmatrix} \rho v_x - \rho v_x^{mesh} \\ \rho v_x^2 + p - \rho v_x v_x^{mesh} \\ \rho v_x v_y - \rho v_y v_x^{mesh} \\ \rho v_x H - \rho E v_x^{mesh} \end{pmatrix} \quad \mathbf{F}_y = \begin{pmatrix} \rho v_y - \rho v_y^{mesh} \\ \rho v_x v_y - \rho v_x v_y^{mesh} \\ \rho v_y^2 + p - \rho v_y v_y^{mesh} \\ \rho v_y H - \rho E v_y^{mesh} \end{pmatrix} \quad (3)$$

where the (u_{mesh}, v_{mesh}) is the mesh velocity. The viscous fluxes \mathbf{F}^V are given by the following expression,

$$\mathbf{F}_x^V = \begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ v_x \tau_{xx} + v_y \tau_{xy} - q_x \end{pmatrix} \quad \mathbf{F}_y^V = \begin{pmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ v_x \tau_{xy} + v_y \tau_{yy} - q_y \end{pmatrix} \quad (4)$$

The viscous stresses are modeled as

$$\begin{aligned} \tau_{xx} &= 2\mu \frac{\partial v_x}{\partial x} - \frac{2}{3}\mu \left(\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} \right) \\ \tau_{yy} &= 2\mu \frac{\partial v_y}{\partial y} - \frac{2}{3}\mu \left(\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} \right) \\ \tau_{xy} &= \mu \left(\frac{\partial v_x}{\partial y} + \frac{\partial v_y}{\partial x} \right) \end{aligned} \quad (5)$$

For incompressible flows, the assumption of incompressibility lead us to a system of equations with the following variables

$$\mathbf{U}(\mathbf{x}, t) = \begin{Bmatrix} 0 \\ v_x \\ v_y \end{Bmatrix} \quad (6)$$

The inviscid fluxes are

$$\mathbf{F}_x = \begin{pmatrix} \rho v_x - \rho v_x^{mesh} \\ \rho v_x^2 + p - \rho v_x v_x^{mesh} \\ \rho v_x v_y - \rho v_y v_x^{mesh} \end{pmatrix} \quad \mathbf{F}_y = \begin{pmatrix} \rho v_y - \rho v_y^{mesh} \\ \rho v_x v_y - \rho v_x v_y^{mesh} \\ \rho v_y^2 + p - \rho v_y v_y^{mesh} \end{pmatrix} \quad (7)$$

The viscous fluxes are given by

$$\mathbf{F}_x^V = \begin{pmatrix} 0 \\ \mu \frac{\partial v_x}{\partial x} \\ \mu \frac{\partial v_y}{\partial x} \end{pmatrix} \quad \mathbf{F}_y^V = \begin{pmatrix} 0 \\ \mu \frac{\partial v_x}{\partial y} \\ \mu \frac{\partial v_y}{\partial y} \end{pmatrix} \quad (8)$$

3. Basic finite volume formulation

The basic finite volume discretization stems from the integral form of Eq. (1) over a control volume Ω_i

$$\int_{\Omega_i} \frac{\partial \mathbf{U}}{\partial t} d\Omega + \int_{\Omega_i} \frac{\partial (\mathbf{F}_x - \mathbf{F}_x^V)}{\partial x} d\Omega + \int_{\Omega_i} \frac{\partial (\mathbf{F}_y - \mathbf{F}_y^V)}{\partial y} d\Omega = \mathbf{0} \quad (9)$$

Using the divergence theorem for the viscous and inviscid fluxes the following expression is obtained

$$\int_{\Omega_I} \frac{\partial \mathbf{U}}{\partial t} d\Omega = \int_{\Gamma_I} (\mathbf{F}^V - \mathbf{F}) \cdot \mathbf{n} d\Gamma \quad (10)$$

where Γ_I is the control volume perimeter and $\mathbf{n} = (n_x, n_y)^T$ is the unitary exterior normal of the contour.

Applying Gauss quadrature to evaluate the integral of the inviscid and viscous flux terms produces the following expression

$$A_I \frac{\partial \mathbf{U}_I}{\partial t} = \sum_{j=1}^{N_f} \sum_{ig=1}^{N_G} [(\mathbf{F}^V - \mathbf{F}) \cdot \hat{\mathbf{n}}_j]_{ig} \mathcal{W}_{ig} \quad (11)$$

where A_I is the area of cell I , \mathbf{U}_I represents the average value of \mathbf{U} over the cell I , N_f is the number of faces of the control volume, N_G represents the number of quadrature points, \mathcal{W}_{ig} is the corresponding quadrature weight for the quadrature point at cell faces, denoted with subscript ig , and $\hat{\mathbf{n}}_j$ is the unitary normal \mathbf{n} times the area of the face j . It has been shown by Chassaing et al. in [5] that when Gauss-based quadrature rules are used the formal order of accuracy of the integration procedure is almost recovered in spite of the non-polynomial form of MLS approximations.

It is critical in the development of robust high order schemes for the Navier–Stokes equations to acknowledge the distinct nature of the inviscid and viscous fluxes [9]. The former is of hyperbolic character, whereas the later is of elliptic character.

The inviscid fluxes are discretized using any suitable numerical fluxes, that is based on the right (+) and left (−) Riemann states of cell edge. In order to solve the Riemann problem the variables at the cell centroid at both sides of the edge need to be approximated to the integration point at the edge, obtaining the states \mathbf{U}^+ and \mathbf{U}^- . The viscous fluxes are evaluated directly at integration points using a continuous approach.

A schematic representation of the reconstructed variables of the inviscid fluxes is shown in Fig. 1.

The solution procedure of the compressible and incompressible set of equations is different. The details can be found in [9,24] for the compressible and [44] for the incompressible Navier Stokes equations.

4. Moving Least-Squares reproducing kernel approximations

In this section we introduce briefly the Moving Least Squares (MLS) approach [25] that is the key ingredient of the new sliding mesh approaches presented in this paper.

We refer the interested reader to [25] for a complete description of the method.

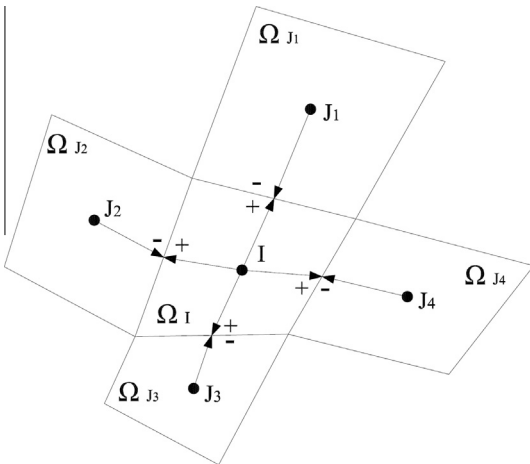


Fig. 1. Reconstructed variables used to evaluate the inviscid fluxes at the edges of the control volume Ω_I .

For the sake of brevity, we only recall that the MLS approximation of a given function $u(\mathbf{x})$ is

$$\hat{u}(\mathbf{x}) = \sum_{j=1}^{n_x} N_j(\mathbf{x}) u_j \quad (12)$$

The approximation is written in terms of the MLS shape functions N^T . In order to compute the MLS shape functions we need a set of neighboring points, n_x , where the variable u_j is known.

The definition of the set of neighbors is a very important point. The stencil should be as compact as possible, and the selection of neighbors must be suitable for general grids. A typical stencil is shown in Fig. 2. It is important to remark that MLS stencils are typically centered around the node. This feature avoids the spatial bias which is often found in patch-based piece-wise polynomial approximations.

The MLS shape functions N^T can be obtained as

$$N^T(\mathbf{x}) = \mathbf{p}^T(\mathbf{x}) \mathbf{M}^{-1}(\mathbf{x}) \mathbf{P}_{\Omega_x} \mathbf{W}(\mathbf{x}) \quad (13)$$

where $\mathbf{p}^T(\mathbf{x})$ is the basis functions, \mathbf{P}_{Ω_x} is defined as a matrix where the basis functions are evaluated at each point of the stencil, $\mathbf{W}(\mathbf{x})$ is a *kernel* or smoothing function and \mathbf{u}_{Ω_x} contains the n_x pointwise values of the function to be reproduced. The moment matrix $\mathbf{M}(\mathbf{x})$ is obtained through the minimization of an error functional [9] and is defined as

$$\mathbf{M}(\mathbf{x}) = \mathbf{P}_{\Omega_x} \mathbf{W}(\mathbf{x}) \mathbf{P}_{\Omega_x}^T \quad (14)$$

The order of MLS approximations is determined by the polynomial basis used in the construction of MLS shape functions. Here we use cubic polynomial basis [8,5].

The *kernel* function $\mathbf{W}(\mathbf{x})$ plays a very important role in the MLS method. It is used to weight the different points that take place in the approximation. A wide variety of *kernel* functions are described in the literature [30,28,29,34], most of them being spline or exponential functions. More details can be found in [35]. In this work the exponential kernel has been used,

$$W_j(x_j, x, s_x) = \frac{e^{-(\frac{d}{c})^2} - e^{-(\frac{d_m}{c})^2}}{1 - e^{-(\frac{d_m}{c})^2}} \quad (15)$$

for $j = 1, \dots, n_x$, where $d = |x_j - x|$, $d_m = 2 \max(|x_j - x|)$.

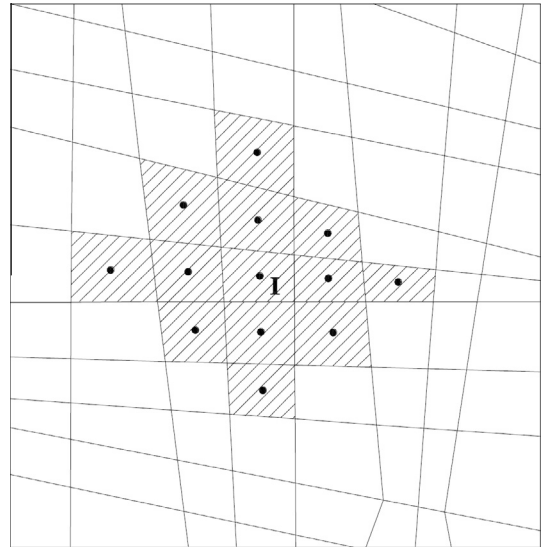


Fig. 2. Typical stencil for interior cells used for cubic MLS approximation centered at cell centroid I .

In Eq. (15) d_m is the *smoothing length*, n_x is the number of neighbors and x is the reference point where the compact support is centered, the coefficient c is defined by $c = \frac{d_m}{s_x}$ and s_x is the shape parameter of the *kernel*. This parameter plays defines the kernel properties and therefore, the properties of the numerical scheme obtained. In Fig. 3 the 1D exponential kernel is plotted for different values of the shape parameter s_x . As it will be shown later, this parameter plays an important role when shock waves are present in the solution. In this work the shape parameter is kept constant as $s_x = 4.5$ for interior cells [35] and the effect of s_x for interface cells will be presented in Section 5.2.2. As far as 2-D applications are concerned, a multidimensional *kernel* can be obtained by means of the product of 1D *kernels*. For instance the 2D exponential *kernel* is defined by the following expression

$$W_j(x_j, y_j, x, y, s_x, s_y) = W_j(x_j, x, s_x)W_j(y_j, y, s_y) \quad (16)$$

The high order approximate derivatives of the field variables $u(\mathbf{x})$ can be expressed in terms of the derivatives of the MLS shape function. So the n -th derivative is defined as

$$\frac{\partial^n \hat{u}}{\partial x^n} = \sum_{j=1}^{n_x} \frac{\partial^n N_j(\mathbf{x})}{\partial x^n} u_j \quad (17)$$

We refer the interested reader to [5,24] for a complete description of the computation of MLS derivatives.

4.1. FV-MLS method

The FV-MLS method is a high-order finite volume method based on MLS approximations. It has been used in this work for the resolution of the set of Eq. (1). We refer to the interested reader to [9,24,44] for a complete description of the method. Here we only focus on the use of MLS for the development of high-order sliding-mesh techniques.

The construction of the stencil is of crucial importance in the context of high-order finite volume methods on unstructured grids [9,39,6]. The number of control volumes must be a compromise between affordable computational cost and solution accuracy [24,35].

The local stencil is constructed by successfully adding neighbor elements sharing a face with elements belonging to the previous layer (Fig. 2). The minimum size of the stencil is dictated by $(p+1)(p+2)/2$ grid nodes for a p th order polynomial basis [9,39]. However, the size of the supporting nodes may be increased in practice in order to enforce the robustness of the numerical method [9].

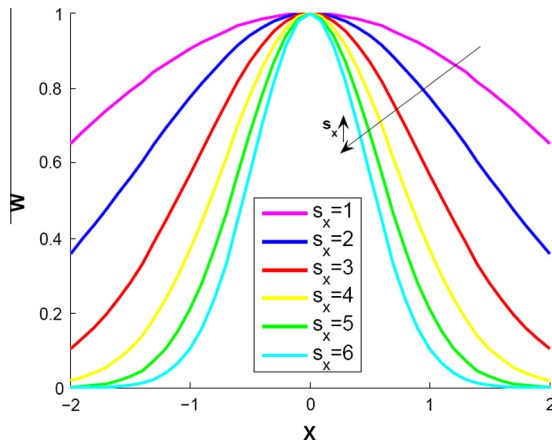


Fig. 3. Representation of the 1D exponential *kernel* for different values of the shape parameter s_x .

It is important to remark that for a stationary mesh the construction of the discretization stencil and the computation of the MLS shape function derivatives are done as a preprocessing step prior to the iterative procedure. As a consequence, the extra computational time relative to the high-order FV-MLS scheme compared to second order scheme is mainly due to the loop over the quadrature points required for the high-order flux integration. If the mesh is not stationary, for example when using the sliding mesh approach, the MLS shape functions need to be recomputed but only in those elements where the stencil has been modified due to mesh displacement.

5. MLS-based sliding mesh

The sliding mesh technique requires two meshed zones in relative motion between them. Even though the methodology presented here is valid for any pair of grids in relative motion, for clarity we focus on the case of one moving grid *sliding* over one fixed grid. This is, for example, the case of turbomachinery, where one of the grids is related to the stator (fixed) and the other is related to the rotor (moving). The sliding takes place on a plane that is called *interface*. Fig. 4 schematically shows this idea. Note that in practice, the interface is composed by two coincident edges. One of them belongs to the fixed grid and the other to the moving mesh.

As the moving grid slides over the fixed grid, the mesh is no longer conformal. Note also that the movement of the mesh is not continuous, since each time step the moving region is displaced a small distance.

In this work we investigate the use of MLS approximations to compute the values required at the interface for the computation of the fluxes in Eq. (11). Hereafter, we call *intersection nodes/cells* to nodes/cells placed at interface between grids in relative motion as shown in Fig. 4. First, the identification of the set of neighbors for each of the intersection cells, namely the computational stencil, must be performed in order to evaluate the MLS-shape functions. To this end, both full stencil and half stencil are considered. The former takes neighboring cells from both grids as shown by shaded triangles in Fig. 5. The latter is simply made with cells of the same grid.

Second, we must develop efficient and robust MLS-based sliding mesh interfaces for the transmission of information for one grid to another. The proposed different techniques belong to the two following sliding mesh families, namely the *intersection-based* and the *interface-halo-cell* approaches.

Therefore, the three high-order approaches, which are presented in Sections 5.1 and 5.2, will be referred hereafter as

1. Full Stencil with Intersections (FS Intersections)
2. Half Stencil with Intersections (HS Intersections)
3. Full Stencil with Halo cell (FS Halo)

5.1. MLS-based sliding mesh with intersections

This approach requires the computation of intersections at the interface. The novelty of this approach is the use of MLS to obtain a high-order accurate reconstruction. We call *main interface* to the interface edge that is part of the moving mesh, and *secondary interface* to the interface edge that belongs to the fixed mesh (see Fig. 6).

First, we identify the main interface edges by performing a loop over the edges of the moving mesh. If one edge is labeled as interface edge, the secondary interface is straightforwardly determined. Once the main and secondary interfaces are determined, intersection nodes can be found.

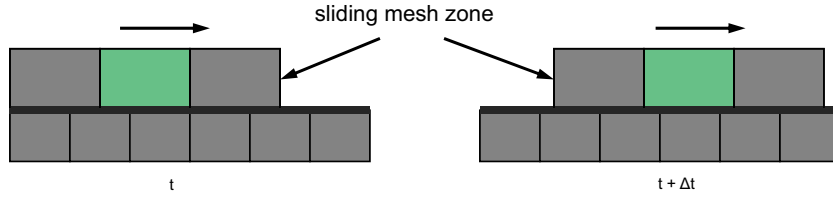


Fig. 4. Schematic illustration of the sliding mesh concept. The top zone slides over the fixed grid. Note the non-conformal grid.

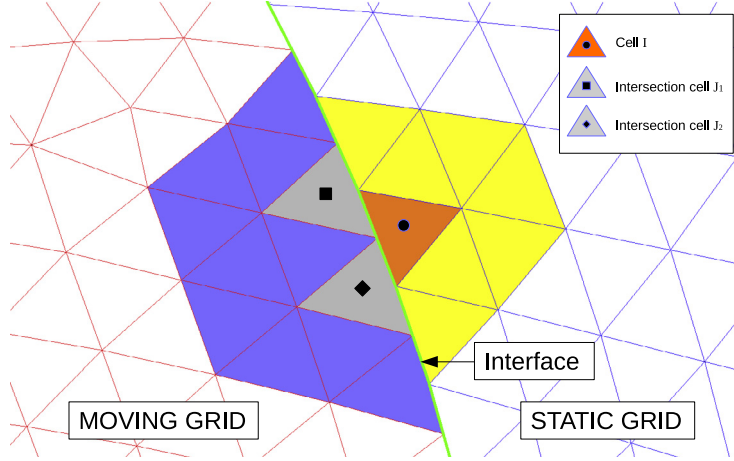


Fig. 5. Schematic illustration of stencil for MLS interpolation at interface.

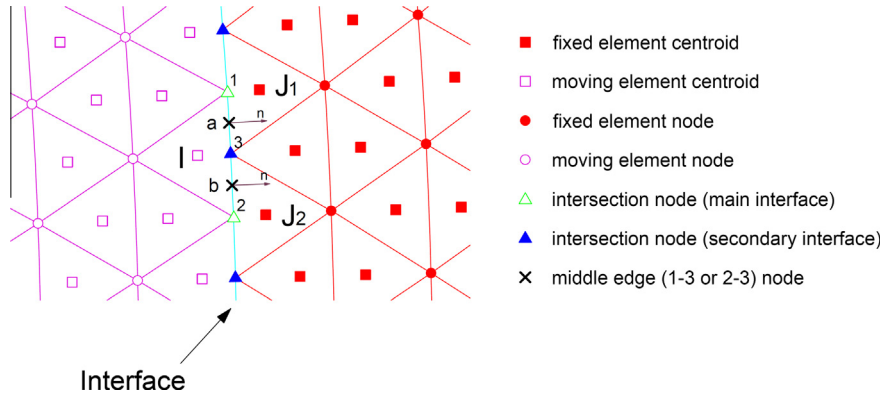


Fig. 6. Schematic representation of intersection nodes.

The exchange of information between the moving domain and the fixed one is performed at both main and secondary interfaces. Since the grid is not conformal at the interface between moving and fixed domains, the corresponding numerical fluxes must be computed for each cells sharing the face of the neighboring interface cell. This procedure is illustrated in Fig. 7 where the flux exchange between cells I/J_1 and cells I/J_2 must be computed.

5.1.1. Full stencil with intersections (FS Intersections)

In the full stencil approach, the stencil of cell I is built using the union of the stencils of the cell I and all the fixed elements of the secondary interface. Therefore, the total stencil of cell I is computed by merging stencils of cell I and those cells having an interface edge coincident with cell I (shaded in orange in Fig. 5). The final stencil of cell I is shown in Fig. 8.

Numerical fluxes at interface are computed using Taylor reconstructions at integration points of each edge. This procedure must be performed each time step. In problems related with rotor/stator

configurations, the location of interface and the numeration of the cells next to it in both domains, fixed and moving is known a priori. The computational cost associated to the evaluation of MLS shape functions at each time-steps is limited since the cells requiring re-computation of the MLS shape functions are those near the interface. This additional cost greatly depends on the grid topology and the level of complexity of the configuration. Thus, the computational times are increased by a maximum factor of 4% for the numerical examples presented in this paper.

5.1.2. Half stencil with intersections (HS Intersections)

In order to avoid both the search of neighbors and the computation of the MLS shape functions at each time step, we propose to employ a biased stencil, taking only into account cells from the grid in which the cell is placed. This procedure is depicted in Fig. 9. Obviously, this half-stencil approach is expected to be less accurate than its full stencil counterpart since the computational stencil of cell I is not centered.

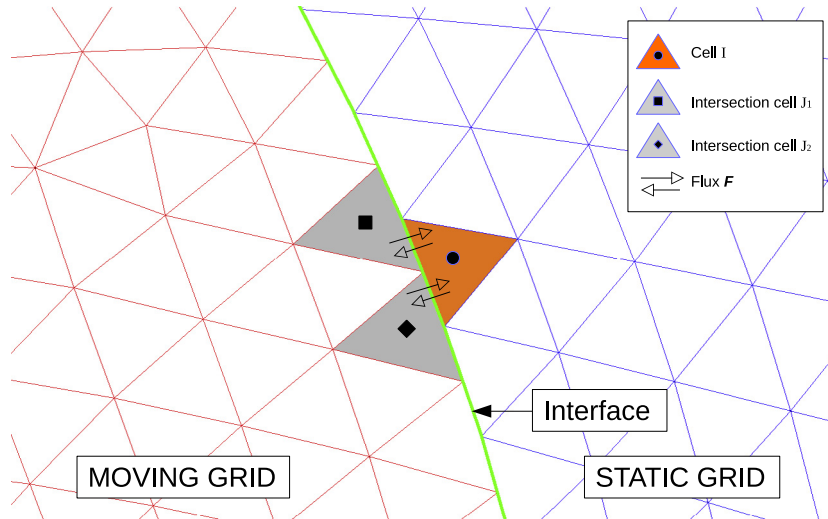


Fig. 7. Schematic representation of the flux exchange for the MLS-based sliding mesh with intersections approach.

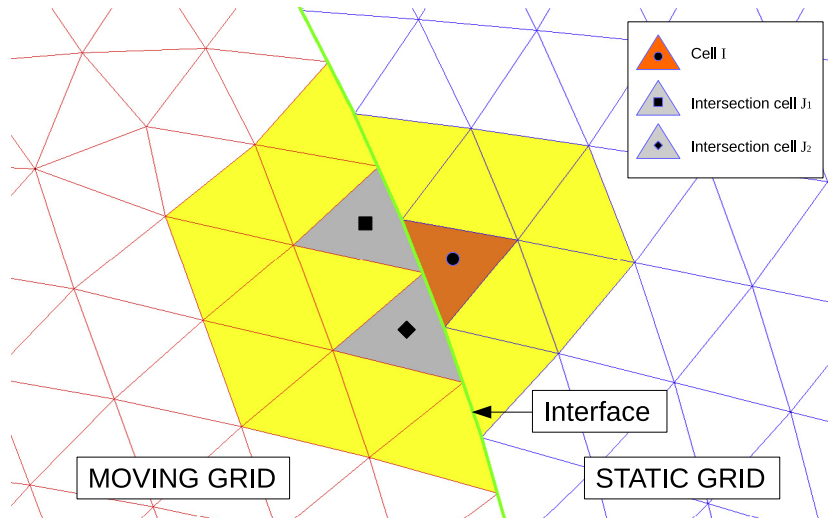


Fig. 8. Schematic representation of the Full Stencil approach.

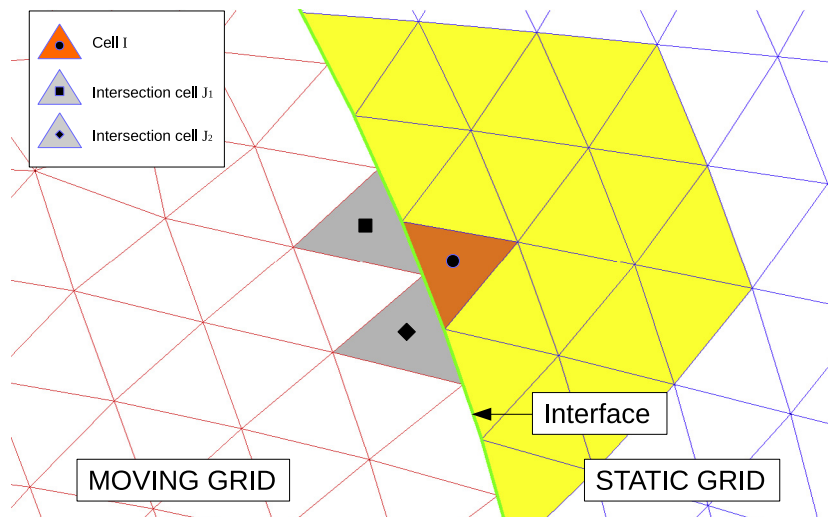


Fig. 9. Schematic representation of the Half Stencil approach.

5.2. Interface-halo-cell sliding mesh

The need for computation of face/edge intersections is one of the drawbacks of previous approach. It therefore introduces additional complexity in the coding and also in the computing time. Here, the Interface-halo-cell stencil is introduced as an alternative to the intersection-based stencils described previously. Let us consider a cell I . We create a *halo* cell in front of this cell, as it is shown in Fig. 10. We call P_H its centroid and A_{P_H} the area of the halo cell. Note that the halo cell is a specular image of cell I . In case of taking into account the curvature of the interface, the areas of I and A_{P_H} will no longer be the same. The value of the variables at P_H is obtained as

$$\mathbf{U}_{P_H} = \frac{1}{A_{P_H}} \int \mathbf{U} dA = \frac{1}{A_{P_H}} \int \sum_{j=1}^{n_x} N_j(\mathbf{x}_{P_H}) U_j dA \quad (18)$$

The value \mathbf{U}^- at the integration point is then computed by a Taylor approximation of \mathbf{U} from P_H and then we compute the flux at cell I as usual. This is schematically presented in Fig. 10.

5.2.1. Halo cell computational stencil

In order to compute the value of the variables \mathbf{U} at the centroid P_H of the halo cell as in (18) we need to compute the MLS shape function $N(\mathbf{x}_{P_H})$ at this point. We also compute the derivatives of the shape function for the Taylor reconstruction of the value of the variables at the interface \mathbf{U}^- . To this end, we need to obtain the stencils for cells I and P_H with cells from both regions (as shown in Fig. 11). Once the stencils are defined, the computation of the MLS shape functions is performed according Eq. (13).

Note that, similar to the intersection-based sliding mesh interfaces, the stencil and shape functions need to be updated each time that the grid moves, but no intersections have to be computed.

The application of the halo cell methodology with other cell-centered high-order technologies is possible. For example, it could be applied to WENO schemes like those proposed in [48,51]. In this case, we need to compute the reconstruction polynomial for the halo cell to define the left/right state of the Riemann problem. A possibility to perform this computation is to define the set of stencils required for the WENO approach by using the most suitable elements from the stencils of the cells overlapped to the halo cell.

5.2.2. Conservation properties of Halo cell interfaces

Although the halo cell approach is not conservative by nature, we aim in this section to demonstrate that the conservation error is within the same order of magnitude than the error in variables, and the convergence order of the conservation error is at least the same than that of the numerical scheme.

To this end, we study the behavior of the Interface-halo-cell sliding mesh formulation when a steady shock is located at the interface. This case is taken from [53]. The flow is assumed to travel from left to right. The conditions on the left and right side of the shock wave are

$$\begin{aligned} \rho_L &= 1, & \rho_R &= 1.8621 \\ u_L &= 1.5, & u_R &= 0.8055 \\ p_L &= 0.71429, & p_R &= 1.7559 \end{aligned} \quad (19)$$

The full computational domain is $0 \leq x \leq 10$ and it is discretized in two regions of 25 elements each. The interface between the two regions is located at $x = 5.0$.

In Fig. 12 we show the results obtained with a single grid for the whole domain and with two grids using the halo cell formulation with full stencil. It is observed that when the parameter of the kernel is less than $s_x = 6$ a spurious solution is obtained. If the kernel parameter is set to $s_x = 6$ for the interface cells, the shock is preserved, as seen in Fig. 12(b). The reason of this behavior lies on the nature of the interpolation. As seen in Fig. 3, when $s_x < 6$ the shape of the kernel is not close enough to the delta function [35], and the value of the variables at the cells of the stencil far from the shock introduce an oscillation that propagates downstream.

In the second test we continue to assess the performance of the Interface-halo-cell sliding mesh formulation. We present an unsteady shock that moves through the interface. This test case corresponds to the Test 1 of [50]. The initial states are

$$\begin{aligned} \rho_L &= 1.0, & \rho_R &= 0.125 \\ u_L &= 0.75, & u_R &= 0.0 \\ p_L &= 1.0, & p_R &= 0.1 \end{aligned} \quad (20)$$

The domain $[0, 1]$ is discretized in two domains with 200 computing cells each and the interface is located at $x = 0.5$. Fig. 13 presents a comparison of the density profile obtained with a single grid and with two different grids with the interface located at $x = 0.5$.

The comparison presented in Fig. 13 shows an excellent agreement between the single grid solution and the halo-cell interface

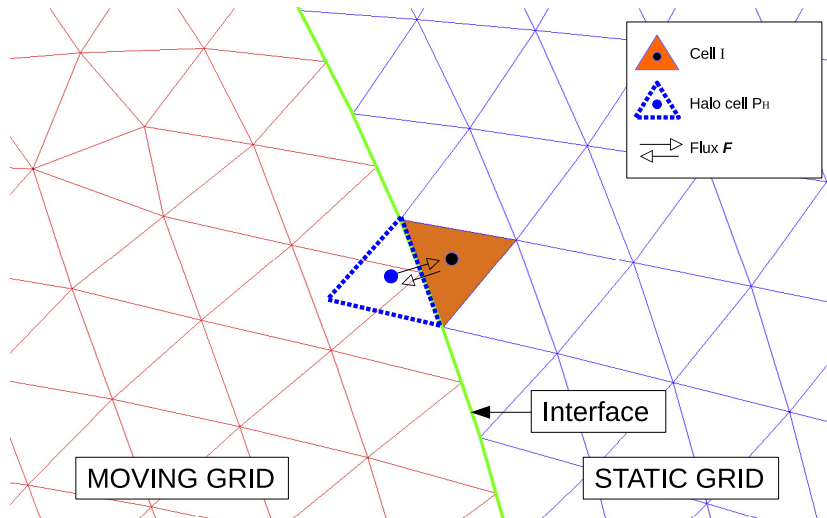


Fig. 10. Schematic representation of Interface-halo-cell sliding mesh. Note that the flux can be computed without the definition of intersections.

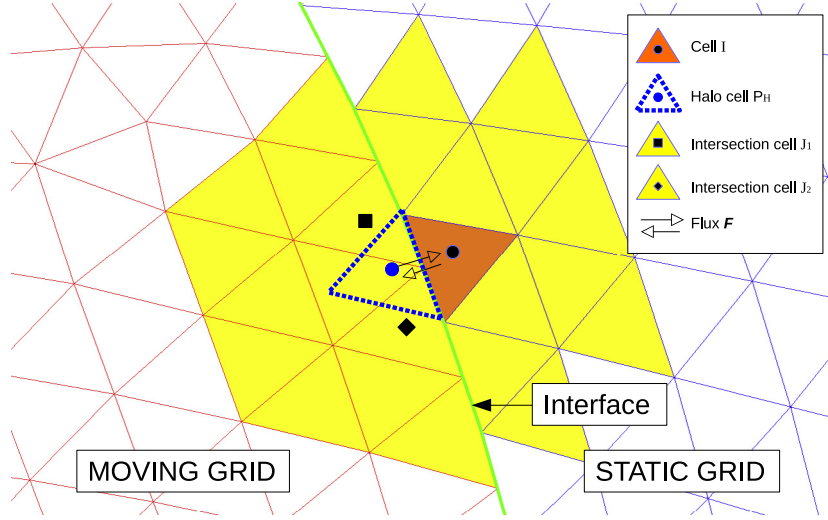


Fig. 11. Schematic representation of Interface-halo-cell sliding mesh with full stencil.

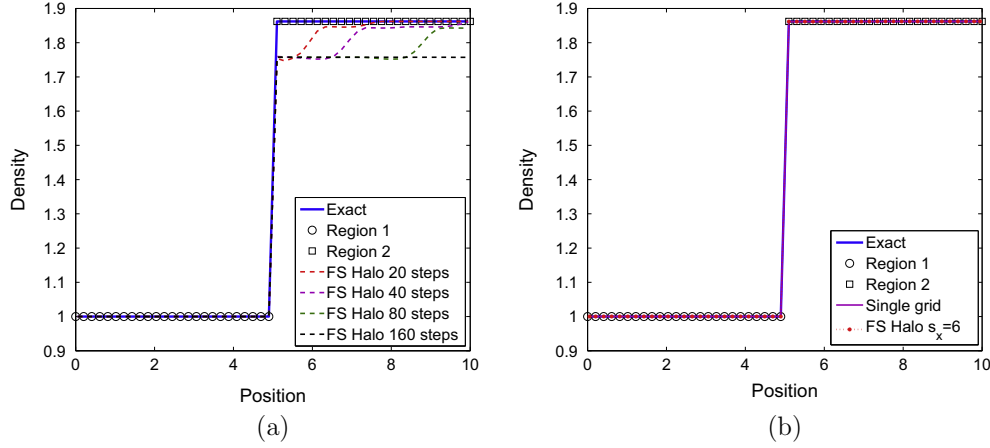


Fig. 12. (a) Time evolution of the numerical results of the 1D steady shock when the FS Halo cell approach is employed with $s_x = 4.5$. The interface is located at $x = 5$. A numerical oscillation propagation can be observed. After 160 steps the stationary state is reached but the shock is not preserved. (b) Comparison of the numerical results of the 1D steady shock between the FS Halo cell approach with $s_x = 6.0$ and when a single mesh grid is employed to discretize the domain. No discrepancies are observed.

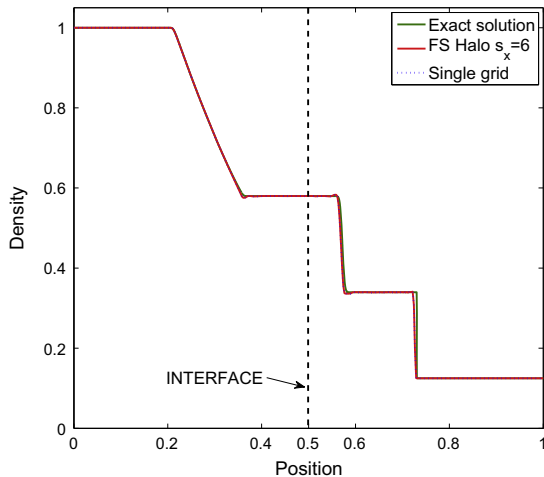


Fig. 13. Numerical solutions at $t = 0.2$ for the 1D Unsteady Shock with 3rd order FV-MLS method. The interface is located at $x = 0.5$.

solution. Note that there is no any visible numerical artifacts at the interface ($x = 0.5$) when the halo cell formulation with full stencil and $s_x = 6$ is used.

6. Numerical examples

This section presents numerical results for several test cases aimed at assessing the accuracy and efficiency of the proposed method for both steady and unsteady flow problems. The first three test cases study the accuracy and the conservation property of the method for non conformal meshes. The last example shows the application of the method to the solution of the incompressible Navier–Stokes equations.

6.1. Ringleb flow

In this 2D example we study the rate of convergence and the conservation error for the different sliding mesh approaches derived in Section 5. This test case is widely used as a benchmark for compressible codes [7,36]. The flow is obtained as a solution of the hodograph equation. The transformation

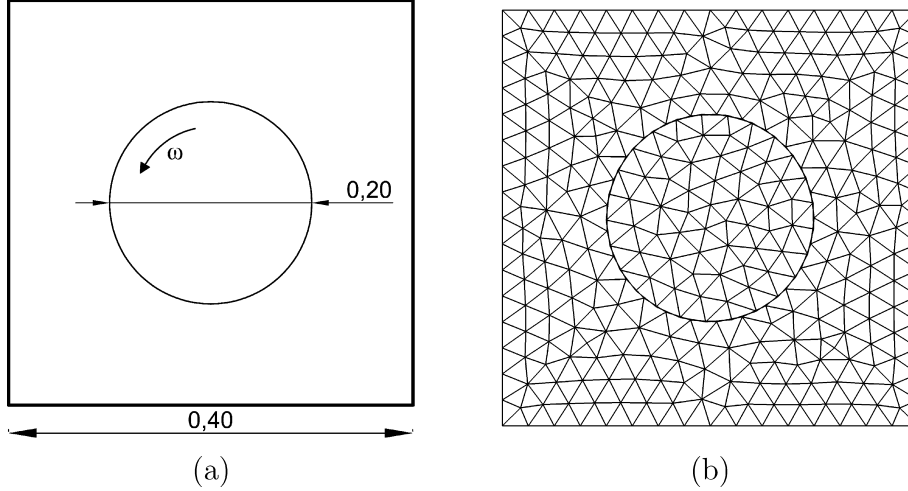


Fig. 14. (a) Geometry description of Ringleb flow problem. (b) Unstructured mesh of 580 triangles. Note the non-conformal mesh at the interface.

Table 1

Accuracy orders, conservation error and L_2 norms of entropy error, total velocity and energy for the Ringleb flow test case for the third-order FV-MLS scheme employing the three different approaches described in this article on non-conformal meshes: Full Stencil Intersections (FSI), Half Stencil Intersections (HSI) and Full Stencil Halo cell (FSH).

Method	Mesh	Entropy error	Order	Conservation error	Order	Velocity error	Order	Energy error	Order
<i>Third order $\omega = 0$ rad/s</i>									
FSI	580	8.44×10^{-8}	–	–	–	1.44×10^{-7}	–	2.16×10^{-7}	–
	2270	9.05×10^{-9}	3.27	–	–	1.75×10^{-8}	3.09	2.40×10^{-8}	3.22
	9044	1.14×10^{-9}	3.00	–	–	2.24×10^{-9}	2.98	2.94×10^{-9}	3.04
HSI	580	9.19×10^{-8}	–	–	–	1.54×10^{-7}	–	2.06×10^{-7}	–
	2270	9.83×10^{-9}	3.28	–	–	1.83×10^{-8}	3.13	2.38×10^{-8}	3.16
	9044	1.27×10^{-9}	2.97	–	–	2.39×10^{-9}	2.94	3.00×10^{-9}	3.00
FSH	580	9.19×10^{-8}	–	2.62×10^{-8}	–	1.47×10^{-7}	–	1.94×10^{-7}	–
	2270	9.29×10^{-9}	3.36	2.03×10^{-9}	3.75	1.82×10^{-8}	3.06	2.32×10^{-8}	3.12
	9044	1.19×10^{-9}	2.98	2.83×10^{-10}	2.85	2.34×10^{-9}	2.97	2.92×10^{-9}	3.00

Table 2

Accuracy orders, conservation error and L_2 norms of entropy error, total velocity and energy for the Ringleb flow test case for the fourth-order FV-MLS scheme employing the three different approaches described in this article on non-conformal meshes: Full Stencil Intersections (FSI), Half Stencil Intersections (HSI) and Full Stencil Halo cell (FSH).

Method	Mesh	Entropy error	Order	Conservation error	Order	Velocity error	Order	Energy error	Order
<i>Fourth order $\omega = 0$ rad/s</i>									
FSI	580	3.76×10^{-9}	–	–	–	8.12×10^{-9}	–	1.39×10^{-8}	–
	2270	2.72×10^{-10}	3.85	–	–	5.61×10^{-10}	3.92	8.32×10^{-10}	4.13
	9044	1.80×10^{-11}	3.93	–	–	3.72×10^{-11}	3.93	6.07×10^{-11}	3.79
HSI	580	1.21×10^{-8}	–	–	–	1.76×10^{-8}	–	2.27×10^{-8}	–
	2270	7.25×10^{-10}	4.12	–	–	1.12×10^{-9}	4.04	1.47×10^{-9}	4.01
	9044	4.00×10^{-11}	4.16	–	–	6.52×10^{-11}	4.11	9.28×10^{-11}	3.99
FSH	580	4.16×10^{-9}	–	3.48×10^{-9}	–	8.31×10^{-9}	–	1.43×10^{-8}	–
	2270	2.66×10^{-10}	4.03	2.97×10^{-11}	6.98	5.43×10^{-10}	4.00	8.58×10^{-10}	4.12
	9044	1.76×10^{-11}	3.93	2.60×10^{-12}	3.53	3.66×10^{-11}	3.90	6.48×10^{-11}	3.74

equations between the Cartesian variables (x, y) and the hodograph variables (V, ϑ) are described in [7]. The square domain $\Omega = [-1.15, -0.75] \times [0.15, 0.55]$ was discretized using four different unstructured mesh with non conformality at the interface of 580, 2270, 9044 and 35,918 triangular elements. Fig. 14 shows the schematic description of the problem and the coarsest grid. The analytical solution is prescribed at the boundaries. In order to check the formal order of accuracy of our numerical solver, we compute the L_2 norms of the entropy, total velocity and energy errors as

$$L_2^{ent} = \sqrt{\frac{1}{\Omega} \int_{\Omega} \left(\frac{p/\rho^\gamma - p_\infty/\rho_\infty^\gamma}{p_\infty/\rho_\infty^\gamma} \right)^2 d\Omega} \quad (21)$$

$$L_2^v = \sqrt{\frac{1}{\Omega} \int_{\Omega} (|\mathbf{v}| - |\mathbf{v}_{exact}|)^2 d\Omega} \quad (22)$$

$$L_2^{ener} = \sqrt{\frac{1}{\Omega} \int_{\Omega} (E - E_{exact})^2 d\Omega} \quad (23)$$

The conservation error is evaluated as the difference of sum of the fluxes from both sides of the interface

Table 3

Accuracy orders, conservation error and L_2 norms of entropy error, total velocity and energy for the Ringleb flow test case with an angular velocity of $\omega = 0.01$ rad/s for the third-order FV-MLS scheme employing the three different approaches described in this article on non-conformal meshes: Full Stencil Intersections (FSI), Half Stencil Intersections (HSI) and Full Stencil Halo cell (FSH).

Method	Mesh	Entropy error	Order	Conservation error	Order	Velocity error	Order	Energy error	Order
<i>Third order $\omega = 0.01$ rad/s</i>									
FSI	580	8.10×10^{-8}	–	–	–	1.40×10^{-7}	–	1.86×10^{-7}	–
	2270	8.96×10^{-9}	3.23	–	–	1.72×10^{-8}	3.08	2.27×10^{-8}	3.09
	9044	1.14×10^{-9}	2.98	–	–	2.24×10^{-9}	2.95	2.90×10^{-9}	2.98
HSI	580	8.76×10^{-8}	–	–	–	1.51×10^{-7}	–	1.92×10^{-7}	–
	2270	1.01×10^{-8}	3.16	–	–	1.88×10^{-8}	3.05	2.32×10^{-8}	3.10
	9044	1.27×10^{-9}	3.01	–	–	2.40×10^{-9}	2.98	2.92×10^{-9}	3.00
FSH	580	8.83×10^{-8}	–	3.46×10^{-8}	–	1.50×10^{-7}	–	1.95×10^{-7}	–
	2270	9.53×10^{-9}	3.36	4.62×10^{-9}	2.95	1.84×10^{-8}	3.07	2.36×10^{-8}	3.10
	9044	1.20×10^{-9}	2.99	3.17×10^{-10}	3.88	2.42×10^{-9}	2.94	2.93×10^{-9}	3.02

Table 4

Accuracy orders, conservation error and L_2 norms of entropy error, total velocity and energy for the Ringleb flow test case with an angular velocity of $\omega = 0.01$ rad/s for the fourth-order FV-MLS scheme employing the three different approaches described in this article on non-conformal meshes: Full Stencil Intersections (FSI), Half Stencil Intersections (HSI) and Full Stencil Halo cell (FSH).

Method	Mesh	Entropy error	Order	Conservation error	Order	Velocity error	Order	Energy error	Order
<i>Fourth order $\omega = 0.01$ rad/s</i>									
FSI	580	3.61×10^{-9}	–	–	–	7.87×10^{-9}	–	1.29×10^{-8}	–
	2270	2.78×10^{-10}	3.76	–	–	6.03×10^{-10}	3.77	9.04×10^{-10}	3.89
	9044	1.92×10^{-11}	3.86	–	–	5.57×10^{-11}	3.45	7.70×10^{-11}	3.56
HSI	580	1.09×10^{-8}	–	–	–	1.63×10^{-8}	–	2.09×10^{-8}	–
	2270	7.25×10^{-10}	3.98	–	–	1.08×10^{-9}	3.98	1.39×10^{-9}	3.97
	9044	4.05×10^{-11}	4.08	–	–	8.11×10^{-11}	3.74	1.06×10^{-10}	3.73
FSH	580	3.89×10^{-9}	–	3.73×10^{-9}	–	7.94×10^{-9}	–	1.38×10^{-8}	–
	2270	2.85×10^{-10}	3.83	1.65×10^{-10}	4.58	5.74×10^{-10}	3.85	9.94×10^{-10}	3.86
	9044	1.99×10^{-11}	3.85	1.12×10^{-12}	3.89	5.35×10^{-11}	3.43	7.76×10^{-11}	3.69

$$L_2^{cv} = \frac{1}{\Gamma_{Interface}} \int \left| \mathbf{F}_e^1 - \mathbf{F}_e^2 \right| d\Gamma_{Interface} \quad (24)$$

where $\Gamma_{Interface}$ denotes the perimeter of the interface, \mathbf{F}_e^1 the convective flux obtained at the interface of region 1 and \mathbf{F}_e^2 is obtained at the interface of region 2.

Tables 1–4 present the L_2 norm of the entropy error, the conservation error and corresponding orders of accuracy for 3rd and 4th order accurate FV-MLS schemes. For validation purpose, we first investigate the results of sliding mesh computations on non conformal static grids (Tables 1 and 2). The three methods show the expected order of accuracy for the L_2 norm of the entropy, total velocity and energy errors. For the third-order scheme all the three approaches give similar results in terms of accuracy, whereas for the fourth-order scheme the HS-Intersection approach is clearly less accurate. The conservation error of the halo-cell approach is below the entropy error, and it keeps the expected order of convergence.

These remarks are confirmed by the plot of the L_2 norm of entropy errors as a function of different mesh resolutions (Fig. 15). All sliding mesh approaches give very similar results for the 3rd order scheme and the less accurate results of HS-Intersection method are clearly visible for 4th order spatial discretizations.

Next we perform the same analysis with a rotational velocity of $\omega = 0.01$ rad/s. The results are plotted in Tables 3 and 4. Note that the accuracy of the sliding mesh methods is not depreciated compared to the static case and similar behavior is observed.

6.2. Vortex convection

The second test case corresponds to the unsteady isentropic vortex convection [45]. In this problem a vortex is convected with the free stream velocity (u_∞, v_∞) through the interface. This problem has an analytical smooth solution which can be subjected to obtain the order of the scheme on a unsteady case. The exact solution is defined as

$$\begin{aligned} \frac{u(x, y, t)}{a_\infty} &= \frac{u_\infty}{a_\infty} - \frac{K}{2\pi a_\infty} \hat{y} e^{\alpha(1-r^2)/2} \\ \frac{v(x, y, t)}{a_\infty} &= \frac{v_\infty}{a_\infty} + \frac{K}{2\pi a_\infty} \hat{x} e^{\alpha(1-r^2)/2} \\ \frac{T(x, y, t)}{T_\infty} &= 1 - \frac{K^2(\gamma-1)}{8\alpha\pi^2 a_\infty^2} e^{\alpha(1-r^2)} \\ \frac{\rho(x, y, t)}{\rho_\infty} &= \left(\frac{T(x, y, t)}{T_\infty} \right)^{\frac{1}{\gamma-1}} \\ \frac{p(x, y, t)}{p_\infty} &= \left(\frac{T(x, y, t)}{T_\infty} \right)^{\frac{\gamma}{\gamma-1}} \end{aligned}$$

where $\hat{x} = x - x_0 - u_\infty t$, $\hat{y} = y - y_0 - v_\infty t$ and $r = \sqrt{\hat{x}^2 + \hat{y}^2}$. In this simulations the chosen parameters are $\alpha = 1$, $\rho_\infty = 1$, $p_\infty = 1$, $(u_\infty, v_\infty) = (2, 2)$, $(x_0, y_0) = (-10, -10)$ and $K = 5$. With this set of parameters the vortex starts at $(x, y) = (-10, -10)$ and at $t = 5$ reach $(x, y) = (0, 0)$.

Fig. 16 shows a schematic description of the problem and an unstructured mesh of 3884 quadrilateral elements employed in the error convergence study.

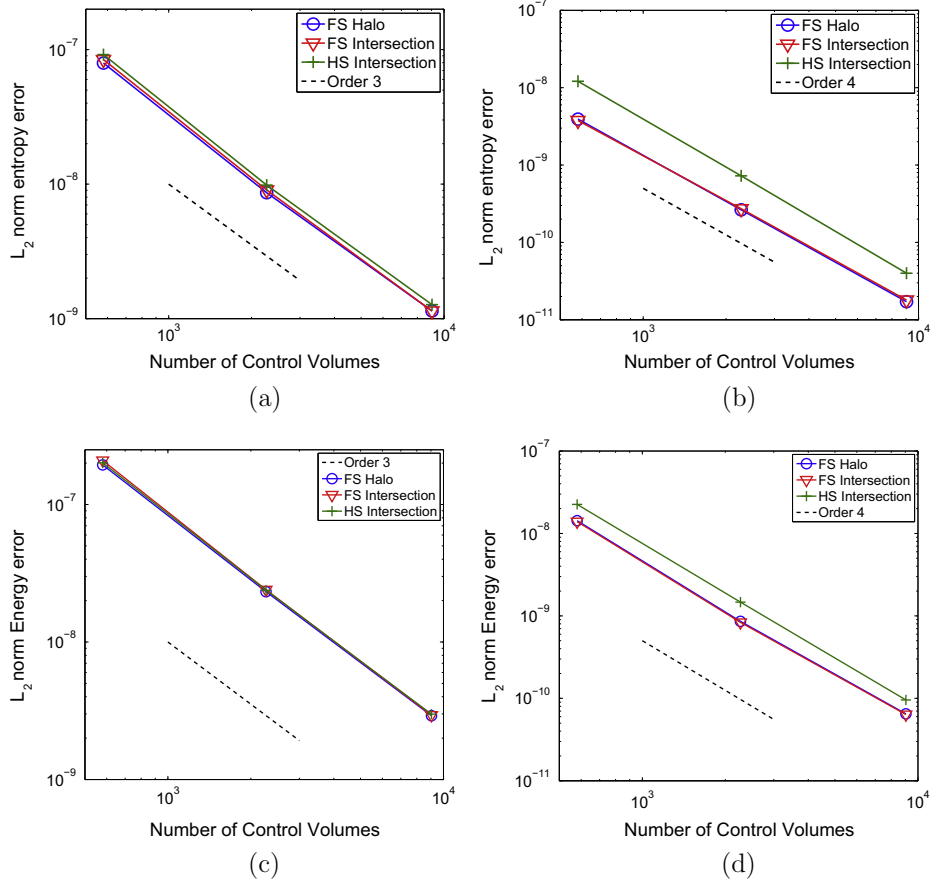


Fig. 15. Ringleb flow ($\omega = 0$). L_2 norms of the entropy error (figures a and b) and of the energy (figures c and d) for the third and fourth FV-MLS schemes using the different sliding mesh schemes proposed. Dashed line denotes the slope of the expected formal order of accuracy.

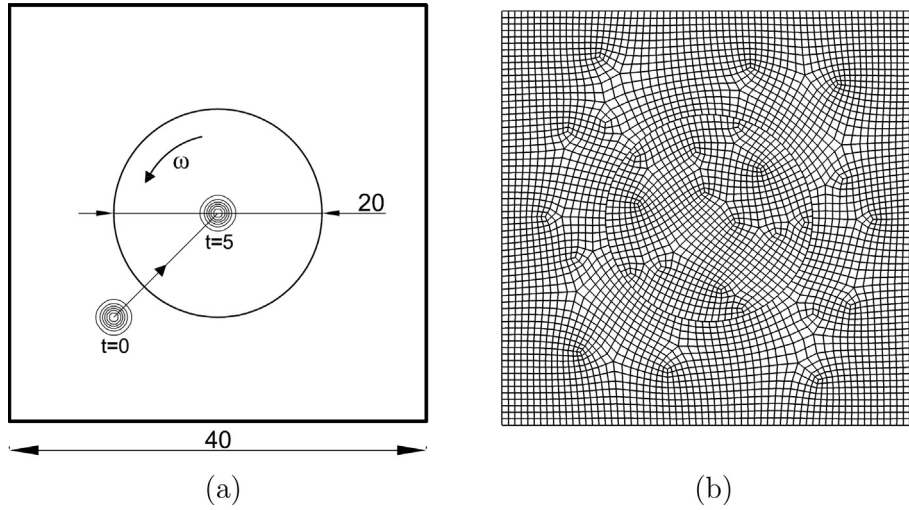


Fig. 16. (a) Geometry description of two dimensional vortex convection problem. (b) Unstructured mesh of 3884 quadrilateral elements.

First, we obtain the L_2 norm of the variables error, conservation error and corresponding orders of accuracy for the case with $\omega = 0$ rad/s and non-conformal meshes. The results are presented in Table 5. Formal third order is reached for the variables error for the three approaches presented in this paper. All methods give very similar results in terms of accuracy.

In Table 6 the results with $\omega = 1.0$ rad/s are showed. It is observed that with rotation the accuracy decreases but the formal order of convergence is not affected.

6.3. Acoustic pulse: Influence of the time step

In this example we aim to analyze the influence of the time step in the accuracy of the FSI, HSI and FSH sliding mesh algorithms proposed.

We compute the propagation of a Gaussian pulse using the non-dimensional Linearized Euler Equations (LEE) [13,47]. We consider zero mean flow and constant mean density and pressure: $\mathbf{U}_0 = [1, 0, 0, 1/\gamma]^T$. The Gaussian pulse is defined as

Table 5

Accuracy orders, conservation error and L_2 norm of variables error for the two dimensional vortex convection for the third-order FV-MLS scheme employing the three different approaches described in this article with an angular velocity of $\omega = 0.0$ rad/s.

Method	Mesh	L_2	Order	Conservation error	Order
<i>Third order $\omega = 0$ rad/s</i>					
FS Intersections	3884	6.03×10^{-2}	–	–	–
	16,407	9.34×10^{-3}	2.59	–	–
	52,136	1.07×10^{-3}	3.74	–	–
HS Intersections	3884	6.27×10^{-2}	–	–	–
	16,407	9.23×10^{-3}	2.66	–	–
	52,136	1.03×10^{-3}	3.80	–	–
FS Halo	3884	6.49×10^{-2}	–	1.07×10^{-1}	–
	16,407	1.08×10^{-2}	2.49	1.34×10^{-2}	2.88
	52,136	1.11×10^{-3}	3.93	1.14×10^{-3}	4.27

Table 6

Accuracy orders, conservation error and L_2 norm of variables error for the two dimensional vortex convection for the third-order FV-MLS scheme employing the three different approaches described in this article with an angular velocity of $\omega = 1.0$ rad/s.

Method	Mesh	L_2	Order	Conservation error	Order
<i>Third order $\omega = 1.0$ rad/s</i>					
FS Intersections	3884	6.26×10^{-2}	–	–	–
	16,407	1.26×10^{-2}	2.23	–	–
	52,136	1.95×10^{-3}	3.22	–	–
HS Intersections	3884	6.10×10^{-2}	–	–	–
	16,407	1.32×10^{-2}	2.13	–	–
	52,136	1.94×10^{-3}	3.31	–	–
FS Halo	3884	6.58×10^{-2}	–	9.74×10^{-2}	–
	16,407	1.31×10^{-2}	2.24	1.52×10^{-2}	2.58
	52,136	1.92×10^{-3}	3.32	1.31×10^{-3}	4.24

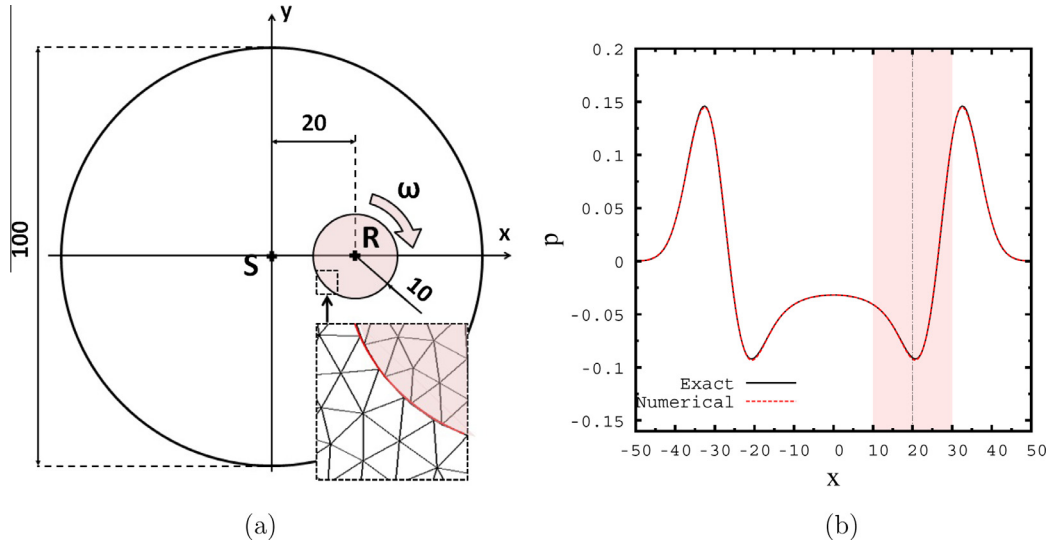


Fig. 17. (a) Geometry description of the domain for the Gaussian pulse propagation problem. (b) Section of the pressure solution along (S, \bar{x}) at $t = 30$ with $N_c = 17,698$, $\Delta T = 0.7$, $\omega = 0.06$. The shaded area corresponds to the moving grid zone.

$$\mathbf{S}(x, y) = \exp\left(-\frac{\ln 2}{b^2} r\right) \times [1 \ 0 \ 0 \ 1]^T \quad (25)$$

where $r = ((x - x_s)^2 + (y - y_s)^2)^{1/2}$ is the radial distance to the location of the center of the pulse (x_s, y_s) . The half-width of the Gaussian

pulse is set at $b = 6$. As in previous cases the computational domain is divided into two grids: a circular fixed grid of diameter $D_F = 100$ centered at point $S(x_S = 0, y_S = 0)$ and a circular sliding grid of diameter $D_R = 20$ centered at $R(x_R = 0, y_R = 20)$ which rotates around its center with a non-dimensional rotational velocity ω_∞ .

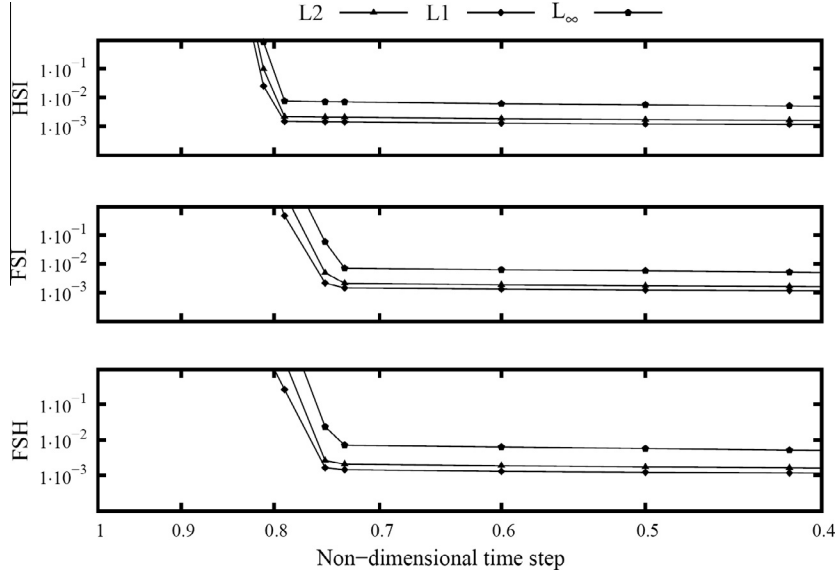


Fig. 18. Error analysis of the proposed MLS-based Sliding Mesh methods for different time steps with a fixed velocity at the sliding mesh interface ($V = 0.6$).

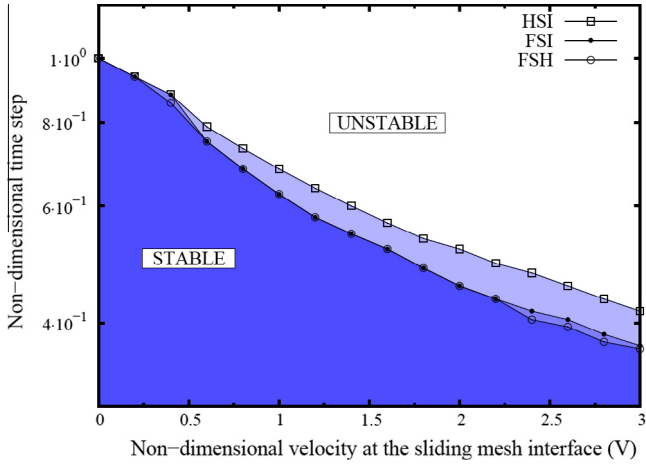


Fig. 19. Maximum time step allowed as a function of the velocity at the interface ($V = \omega R$) for each of the proposed sliding mesh methods.

The source is located at S . This configuration is schematically shown in Fig. 17(a). The analytical solution of this problem is given in [47]. A fourth-order low storage explicit Runge–Kutta (LSERK) method introduced by [4] is used. In order to perform a non-axisymmetric benchmark, the moving grid is not centered on the source ($y_S \neq y_R$).

No numerical artifacts are observed at the interface, as can be seen in Fig. 17(b). Fig. 18 shows the L_1 , L_2 and L_∞ norms of the error in the pressure for each method at a given non-dimensional rotational velocity $\omega = 0.06$ using different time steps. We also evaluate the maximum time step allowed for different values of the velocity at the sliding mesh interface ($V = \omega R$) ranging from 0 to $3c_\infty$, where c_∞ is the non-dimensional speed of sound. Notice that this speed range includes most of the turbomachinery applications (see Fig. 19).

Fig. 18 shows that the error using the half stencil approach is the smallest of the three approaches. This effect was already observed for some of the meshes tested in the vortex convection case. In our opinion, this is due to the directionality of the pulse propagation (or vortex convection), and this behavior is not expected for general cases. On the other hand, the results obtained with the FSI and FSH are very similar.

6.4. Supersonic flow over a cylinder

In this section we present the results of the numerical method applied to the simulation of the inviscid supersonic flow over a cylinder. The aim of this problem is to analyze the performance of the Interface Halo cell Sliding Mesh approach with supersonic flows and to study the suitability of the halo formulation to deal with shocks through the interface in a multidimensional problem. The problem setup is presented in Fig. 20.

The computational domain is discretized with a structured non-conformal mesh of 7200 quadrilateral elements. Fig. 21 shows the mesh and a close view of the interface.

The freestream Mach number is $M = 3.0$. Following [37] we have used a shock detector based on MLS and the limiter of Van Albada [52] was used.

The benchmark parameters for this test case are the pressure coefficient C_p , the normalized stagnation pressure p_0 and the stand-off distance normalized by the diameter of the cylinder. The stand-off distance is defined in Fig. 22.

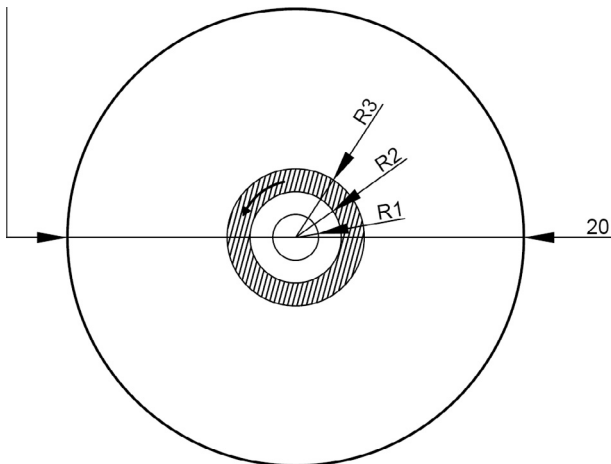


Fig. 20. Geometry description of the supersonic flow over a cylinder. The shaded ring denotes the rotational zone.

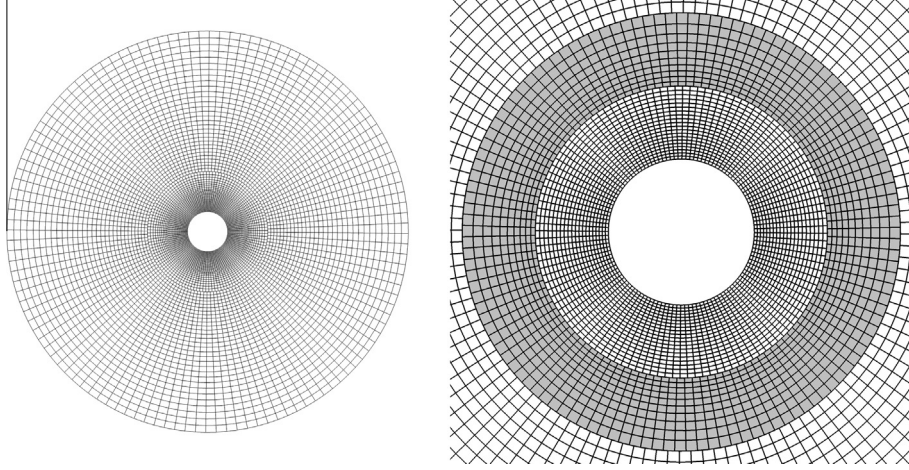


Fig. 21. Structured non-conformal mesh of 7200 quadrilateral elements and closer view to the cylinder and the interfaces.

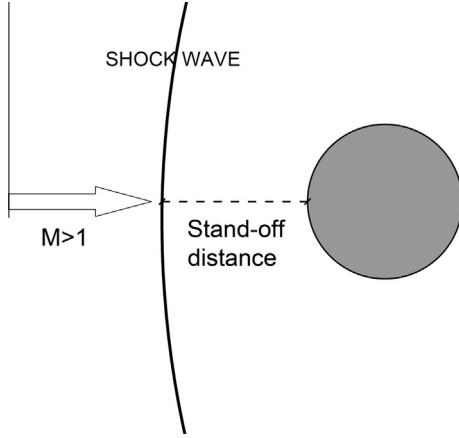


Fig. 22. Schematic representation of the stand-off distance.

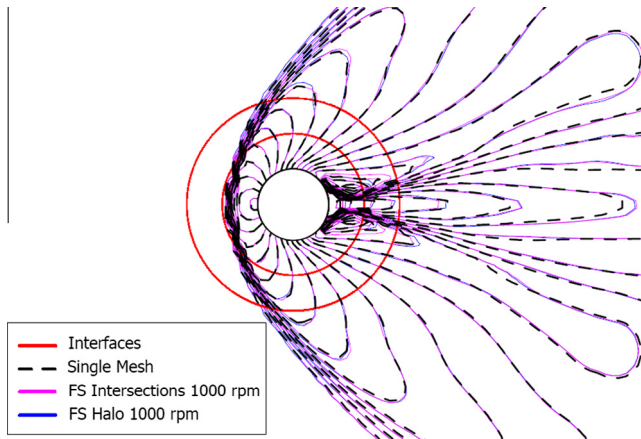


Fig. 23. Comparison of Mach field for the supersonic flow over a cylinder. The solutions are obtained using third-order FV-MLS scheme. Dashed line denotes the solution obtained with a single mesh, purple line is obtained with FS Intersections approach and blue line refers to the solution obtained when the FS Halo approach is employed. The interfaces are highlighted in red. Note that blue and purple lines are practically coincident. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The numerical results obtained for Mach number isolines with sliding mesh FS intersections and FS halo cell are plotted in Fig. 23 for a rotational velocity of 1000 rpm. In order to compare

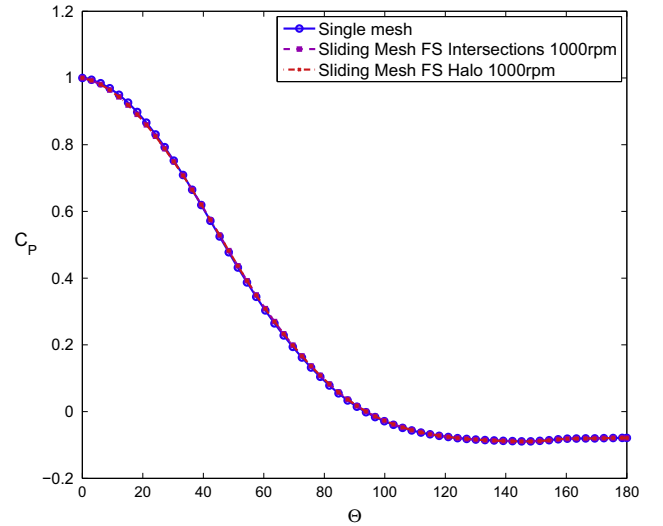


Fig. 24. Comparison of the C_p distribution the supersonic flow over a cylinder. The solutions are obtained using third-order FV-MLS scheme.

the results we have computed the case on a conformal single mesh with no interface. Almost no differences can be observed between the different solutions.

In Fig. 24 the pressure coefficient around the cylinder is compared for the different approaches. Again no differences are observed. In Table 7 we show the results obtained for the normalized stagnation pressure p_0 and the stand-off distance normalized by the diameter of the cylinder. The sliding-mesh computations give the same results. These values are in very good agreement with the ones obtained for a single grid and to the reference solution [27].

6.5. Three bladed cross-flow turbine

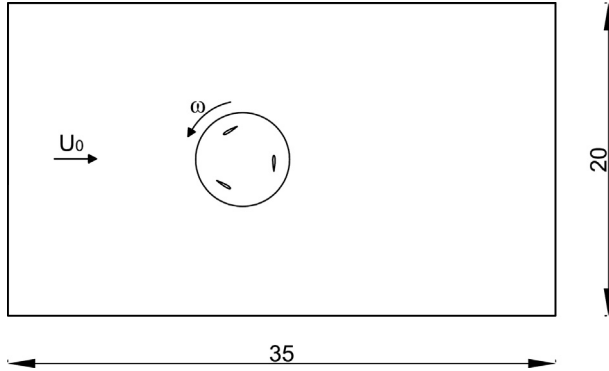
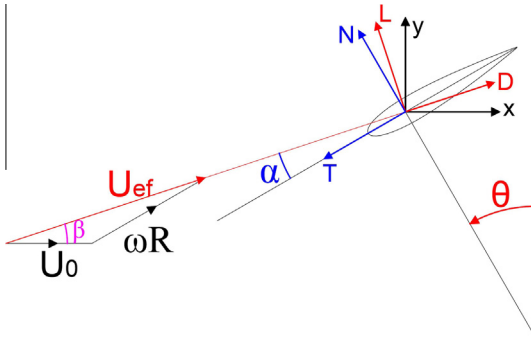
In order to demonstrate the capabilities of our higher-order sliding-mesh interfaces to deal with viscous flows on complex configurations, we compute the unsteady incompressible flow through a cross-flow turbines. Such cross-flow turbine configuration leads to the occurrence of complex flow phenomena, such as blade-vortex interactions.

The problem setup considered in the present work was defined by Ferrer and Willden in [12]. The incompressible Navier-Stokes equations are solved using the SIMPLE algorithm with a collocated

Table 7

Comparison of the normalized stagnation pressure p_0 and the stand-off distance obtained using a single mesh and the three sliding mesh methods (present computations were performed using a 3rd order FV-MLS scheme).

Method	$p_0/(p_0)_\infty$	Stand-off distance/D
Single mesh	0.327	0.405
Sliding Mesh FS Halo 0 rpm	0.324	0.407
Sliding Mesh FS Halo 1000 rpm	0.324	0.408
Sliding Mesh FS Intersections 1000 rpm	0.324	0.408
Reference solution [27]	0.328	–

**Fig. 25.** Basic configuration of the three bladed cross-flow turbine.**Fig. 26.** Diagram of velocities, angles and forces on a single blade.

grid arrangement. We refer the interested reader to [44] for a complete description of the methodology. Here, we have used the Interface-halo-cell approach. The basic configuration of the problem is shown in Fig. 25. The turbine is formed with three NACA0015 airfoil of chord $c = 1$. The blades are disposed with a relative angle of 120° between them and a radial distance of $R = 2c$ from the center of rotation.

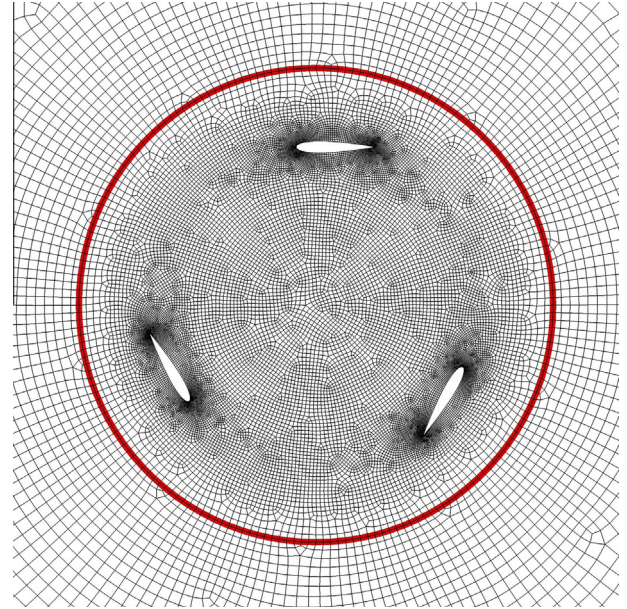
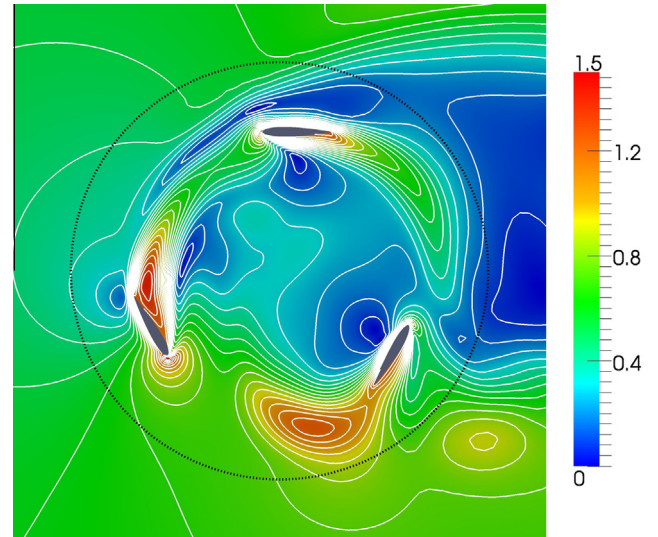
The free-stream velocity is $U_0 = 0.5$. The Reynolds number is defined as $Re = \frac{U_0 c}{\nu} = 50$ and the rotational velocity $\omega = 0.5$.

The tip speed ratio which is, defined as $\lambda = \omega R / U_0$, is equal to $\lambda = 2$ in this work.

The angle of attack α corresponds to the angle between the chord line and the direction of the relative flow direction. Since the turbine rotates α varies throughout the rotation cycle as shown in Fig. 26. The angle of attack can be expressed as function of the tip speed ratio and the angular location θ as [32]

$$\alpha = \tan^{-1} \left(\frac{\sin \theta}{\cos \theta + \lambda} \right) \quad (26)$$

The computational domain is discretized with an unstructured mesh of 26234 quadrilateral elements. Note that with this number of elements, the spatial resolution of the mesh is similar to the one in [12]. A close view of the computational grid near the rotating grid is presented in Fig. 27.

**Fig. 27.** Close view of the unstructured mesh around the cross flow turbine. The red line denotes the interface. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)**Fig. 28.** Velocity field and contours around the three bladed cross-flow turbine. The solution is obtained with a third-order FV-MLS method. The solution is obtained after three rotation cycles, $\theta = 1080^\circ$.

The force on each blade is computed in Cartesian coordinates as

$$\mathbf{f} = \begin{Bmatrix} f_x \\ f_y \end{Bmatrix} = \oint (p\mathbf{n} - \nu(\nabla\mathbf{U} \cdot \mathbf{n}))d\Gamma \quad (27)$$

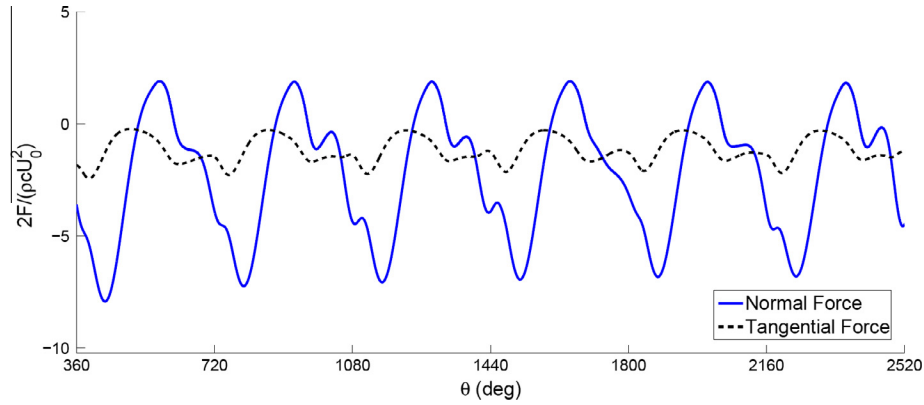


Fig. 29. Normalized tangential and normal forces against azimuth for one blade.

where \mathbf{n} is the outward pointing normal at each blade. Once the forces are obtained in Cartesian coordinates, they can be expressed on a normal-tangential frame on each blade as

$$f_N = f_y \cos \theta - f_x \sin \theta \quad (28)$$

$$f_T = -f_x \cos \theta - f_y \sin \theta \quad (29)$$

The numerical solution obtained after three rotation cycles is plotted in Fig. 28 in terms of the velocity magnitude. We observe no numerical artifacts near the interface and the velocity isolines are perfectly smooth across the non conformal grids.

Fig. 29 compare the normalized tangential and normal forces against the angular rotation θ for one of the blades of the turbine. Results are in good agreement with those obtained by Ferrer [11] using 3rd order Discontinuous Galerkin method.

7. Conclusions

In this paper we have presented a new high-order-preserving sliding-mesh methodology based on Moving Least Squares approximants. Two distinct approaches were considered, namely the intersection-based and the *halo cell* sliding mesh methods. The latter is more flexible since it avoids the computation of intersections. As a drawback, it does not conserves mass from a theoretical point of view. However numerical results show that conservation errors are within the order of magnitude of the intersection-based sliding mesh and the rate of convergence is not affected. The accuracy and robustness of the new methodology has been investigated using various numerical test for both inviscid compressible and viscous incompressible flows. Numerical results have shown that the proposed high-resolution sliding-mesh methods are able to preserve the formal order of accuracy of the high-order spatial discretization scheme. One clear advantage of the present methodology is that the same high-order discretization scheme is used through the whole computational domain involving both static and moving grids, naturally avoiding numerical artifacts.

Acknowledgments

This work has been partially supported by the *Ministerio de Ciencia e Innovación* (#DPI2010-16496) and the *Ministerio de Economía y Competitividad* (#DPI2012-33622) of the Spanish Government, the *Consellería de Educación e Ordenación Universitaria* of the *Xunta de Galicia* (grants #CN2011/002 and #GRC2014/039), and the *Universidade da Coruña* (grant INDITEX-UDC).

References

- [1] Bazilevs Y, Hsu MC, Akkerman I, Wright S, Takizawa K, Henicke B, et al. 3D simulation of wind turbine rotors at full scale. Part I: geometry modeling and aerodynamics. *Int J Numer Methods Fluids* 2011;65:207–35.
- [2] Bazilevs Y, Korobenko A, Deng X, Yan J. Novel structural modeling and mesh moving techniques for advanced fluid-structure interaction simulation of wind turbines. *Int J Numer Methods Fluids* 2014. <http://dx.doi.org/10.1002/nme.4738>.
- [3] Berger M, LeVeque R. An adaptive cartesian mesh algorithm for the Euler equations in arbitrary geometries. *AIAA Paper* 89-1930-CP;1989.
- [4] Carpenter MH, Kennedy CA. Fourth-order 2N-storage Runge–Kutta schemes. *NASA Technical Memorandum*, 109112; 1994.
- [5] Chassaing JC, Khelladi S, Nogueira X. Accuracy assessment of a high-order Moving Least Squares finite volume method for compressible flows. *Comput Fluids* 2013;71:41–53.
- [6] Chenoweth SKM, Soria J, Ooi A. A singularity-avoiding Moving-Least-Squares scheme for two-dimensional unstructured meshes. *J Comput Phys* 2009;228:5592–619.
- [7] Chiochia G. Exact solutions to transonic and supersonic flows. *Technical Report AR-211*; 1985.
- [8] Cueto-Felgueroso L, Colominas I, Fe J, Navarrina F, Casteleiro M. High-order finite volume schemes on unstructured grids using Moving Least Squares reconstruction. Application to shallow water dynamics. *Int J Numer Methods Eng* 2006;65:295–331.
- [9] Cueto-Felgueroso L, Colominas I, Nogueira X, Navarrina F, Casteleiro M. Finite-volume solvers and Moving Least Squares approximations for the compressible Navier–Stokes equations on unstructured grids. *Comput Methods Appl Mech Eng* 2007;196:4712–36.
- [10] Erdos JL, Alznert E, McNally W. Numerical solution of periodic transonic flow through a fan stage. *AIAA J* 1977;15:1559–68.
- [11] Ferrer E. A high order discontinuous Galerkin Fourier incompressible 3d Navier–Stokes solver with rotating sliding meshes for simulating cross-flow turbines, PhD. University of Oxford; 2012.
- [12] Ferrer E, Willden RHJ. A high order discontinuous Galerkin Fourier incompressible 3d Navier–Stokes solver with rotating sliding meshes. *J Comput Phys* 2012;231:7037–56.
- [13] Foulquié C, Khelladi S, Ramirez L, Nogueira X, Deligant M, Mardjono J, et al. Toward a high-order preserving sliding-mesh approach for computational aeroacoustics in subsonic turbomachinery. In: *European turbomachinery conference, ETC2015-258*, Madrid; 2015.
- [14] Francois B, Costes M, Dufour G. Comparison of Chimera and sliding mesh techniques for unsteady simulations of counter rotating open-rotors. *ISABE-2011-1231*; 2011.
- [15] Gourdain N, Montagnac M, Wlassow F, Gazeix M. High-performance computing to simulate large-scale industrial flows in multistage compressors. *Int J High Perform Comput Appl* 2010;24:429–43.
- [16] He L. An euler solution for unsteady flows around oscillating blades. *J Turbomach* 1990;114:714–22.
- [17] He L. Method of simulating unsteady turbomachinery flows with multiple perturbations. *AIAA J* 1992;30:2730–5.
- [18] Janajreh I, Qudaih R, Talab I, Ghenai C. Aerodynamic flow simulation of wind turbine: downwind versus upwind configuration. *Energy Convers Manage* 2010;51:1656–63.
- [19] Jaohindy P, Ennamiri H, Garde F, Bastide A. Numerical investigation of airflow through a Savonius rotor. *Wind Energy* 2014;17:853–68.
- [20] Jaohindy P, McTavish S, Garde F, Bastide A. An analysis of the transient forces acting on Savonius rotors with different aspect ratios. *Renew Energy* 2013;55:286–95.
- [21] Kacprzak K, Liskiewicz G, Sobczak K. Numerical investigation of conventional and modified Savonius wind turbines. *Renew Energy* 2013;60:578–85.

- [22] Kang C, Liu H, Yang X. Review of fluid dynamics aspects of Savonius-rotor-based vertical-axis wind rotors. *Renew Sustain Energy Rev* 2014;33:499–508.
- [23] Kang Y-S, Sohn D, Kim JH, Kim H-G, Im S. A sliding mesh technique for the finite element simulation of fluid-solid interaction problems by using variable-node elements. *Comput Fluids* 2014;130:91–104.
- [24] Khelladi S, Nogueira X, Bakir F, Colominas I. Toward a higher-order unsteady finite volume solver based on reproducing kernel particle method. *Comput Methods Appl Mech Eng* 2011;200:2348–62.
- [25] Lancaster P, Salkauskas K. Pseudo-divergence-free element free Galerkin method for incompressible fluid flow. *Comput Methods Appl Mech Eng* 2004;193:1119–36.
- [26] Lanzafame R, Mauro S, Messina M. 2D CFD modeling of h-Darrieus wind turbines using a transition turbulence model. *Energy Procedia* 2014;45:131–40.
- [27] Liepmann HW, Roshko A. *Elements of gasdynamics*. Dover; 2002.
- [28] Liu GR, Gu YT. A point interpolation method. In: *Proc 4th Asia-Pacific conference on computational mechanics*. Singapore; 1999. p. 1009–14.
- [29] Liu GR, Gu YT, Dai KY. Assessment and applications of point interpolation methods for Computational Mechanics. *Int J Numer Methods Eng* 2004;59:1373–97.
- [30] Liu GR, Liu MB. *Smoothed particle hydrodynamics. A meshfree particle method*. Singapore: World Scientific Publishing; 2003.
- [31] McNaughton J, Afgan I, Apsley DD, Rolfo S, Stallard T, Stansby PK. A simple sliding-mesh interface procedure and its application to the CFD simulation of a tidal-stream turbine. *Int J Numer Methods Fluids* 2014;74:250–69.
- [32] McNaughton J, Billard F, Revell A. Turbulence modelling of low Reynolds number flow effects around a vertical axis turbine at a range of tip-speed ratios. *J Fluids Struct* 2014;47:124–38.
- [33] Moon YJ, Cho Y, Nam H-S. Computation of unsteady viscous flow and aeroacoustic noise of cross flow fans. *Comput Fluids* 2003;32:995–1015.
- [34] Most T, Bucher C. New concepts for Moving Least squares: an interpolation non-singular weighting function and weighted nodal least squares. *Eng Anal Boundary Elem* 2008;32:461–70.
- [35] Nogueira X, Colominas I, Cueto-Felgueroso L, Khelladi S. On the simulation of wave propagation with a higher-order finite volume scheme based on reproducing kernel methods. *Comput Methods Appl Mech Eng* 2010;199:1471–90.
- [36] Nogueira X, Cueto-Felgueroso L, Colominas I, Gomez H, Navarrina F, Casteleiro M. On the accuracy of finite volume and discontinuous Galerkin discretizations for compressible flow on unstructured grids. *Int J Numer Methods Eng* 2009;78:1553–84.
- [37] Nogueira X, Cueto-Felgueroso L, Colominas I, Navarrina F, Casteleiro M. A new shock-capturing technique based on Moving Least Squares for higher-order numerical schemes on unstructured grids. *Comput Methods Appl Mech Eng* 2010;199:2544–58.
- [38] Nogueira X, Khelladi S, Colominas I, Cueto-Felgueroso L, París J, Gómez H. High-resolution finite volume methods on unstructured grids for turbulence and aeroacoustics. *Arch Comput Methods Eng* 2011;18:315–40.
- [39] Ollivier-Gooch C, Nejat A, Michalak K. Obtaining and verifying high-order unstructured finite volume solution to the Euler equations. *AIAA J* 2009;47:2105–20.
- [40] Peskin CS. Flow patterns around heart valves: a numerical method. *J Comput Phys* 1972;10:252–71.
- [41] Rai M. A conservative treatment of zonal boundaries for Euler equation calculations. *J Comput Phys* 1986;62:472–503.
- [42] Rai M. A relaxation approach to patched-grid calculations with the Euler equations. *J Comput Phys* 1986;66:99–131.
- [43] Rai M. Navier–Stokes simulations of rotor-stator interaction using patched and overlaid grids. *J Comput Phys* 1987;3:387–96.
- [44] Ramirez L, Nogueira X, Khelladi S, Chassaing JC, Colominas I. A new higher-order finite volume method based on Moving Least Squares for the resolution of the incompressible Navier–Stokes equations on unstructured grids. *Comput Methods Appl Mech Eng* 2014;278:883–901.
- [45] Shu CW. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. *Lecture Notes in Mathematics*. Springer; 1998.
- [46] Steijl R, Barakos G. Sliding mesh algorithm for CFD analysis of helicopter rotor/fuselage aerodynamics. *Int J Numer Methods Fluids* 2008;58:527–49.
- [47] Tam CKW, Web JC. Dispersion-relation-preserving finite difference schemes for computational aeroacoustics. *J Comput Phys* 1993;107:262–81.
- [48] Titarev VA, Tsoutsanis P, Drikakis D. WENO schemes for mixed-element unstructured meshes. *Commun Comput Phys* 2010;8:585–609.
- [49] Toffolo A. On the theoretical link between design parameters and performance in cross-flow fans: a numerical and experimental study. *Comput Fluids* 2005;34:49–66.
- [50] Toro EF. *Riemann solvers and numerical methods for fluid dynamics. A practical introduction*. Springer; 1999.
- [51] Tsoutsanis P, Titarev VA, Drikakis D. WENO schemes on arbitrary mixed-element unstructured meshes in three space dimensions. *J Comput Phys* 2011;230:1585–601.
- [52] Van Albada GD, Van Leer B, Roberts WW. A comparative study of computational methods in cosmic gas dynamics. *Astron Astrophys* 1982;108:76–84.
- [53] Wang ZJ, Hariharan N, Chen R. Recent development on the conservation property of Chimera. *Int J Comput Fluid Dyn* 2001;15:265–78.
- [54] Ye T, Mittal R, Udaykumar HS, Shyy W. An accurate cartesian grid method for viscous incompressible flows with complex immersed boundaries. *J Comput Phys* 1999;156:209–49.